

Workshop N° 2

MEZIOUNI Reda 07/01/2021 1. Tester si Git existe déjà au niveau de la machine.

```
redkhay@vm:~$ git --version
git version 2.30.0
redkhay@vm:~$ |
```

2. Configurer Git et GitHub.

```
redkhay@vm:~$ git config --global user.name "RedaMeziouni"
redkhay@vm:~$ git config --global user.email "meziounir@gmail.com"
redkhay@vm:~$ git config --global user.name
RedaMeziouni
redkhay@vm:~$ git config --global user.email
meziounir@gmail.com
redkhay@vm:~$ |
```

3. Donnant un peu de vie à notre code Git

```
redkhay@vm:~$ git config --global color.ui auto redkhay@vm:~$
```

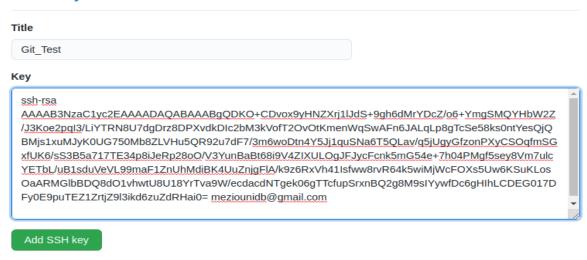
4. Changé la branche par défaut de « master » « main »

```
redkhay@vm:~$ git config --global init.defaultBranch main redkhay@vm:~$
```

5. Créer un clé SSH et l'afficher grâce a la commande « cat »

6. lier à GitHub par clé SSH

SSH keys / Add new



7. La création d'une nouveau repo sur GitHub et le cloner au niveau de ma machine.



```
redkhay@vm:~$ ls

Desktop Documents Downloads Music Pictures Public snap Templates Videos redkhay@vm:~$ mkdir Git redkhay@vm:~$ cd Git/ redkhay@vm:~\Git\$ git clone git@github.com:RedaMeziouni/test_git.git Cloning into 'test_git'... warning: You appear to have cloned an empty repository. redkhay@vm:~/Git\$ |
```

```
8. redkhay@vm:~/Git$ ls
test_git
redkhay@vm:~/Git$ cd test_git/
redkhay@vm:~/Git/test_git$ git remote -v
origin git@github.com:RedaMeziouni/test_git.git (fetch)
origin git@github.com:RedaMeziouni/test_git.git (push)
redkhay@vm:~/Git/test_git$
```

la commande Git remote a pour le rôle de visualiser les serveurs distants que vous avez enregistrés. Elle liste les noms des différentes références distantes que vous avez spécifiées. Si vous avez cloné un dépôt, vous devriez au moins voir l'origine origin — c'est-à-dire le nom par défaut que Git donne au serveur à partir duquel j'ai cloner.

9. Création du fichier README.md

Description:

La commande Git status affiche la liste des fichiers modifiés ainsi que les fichiers qui doivent encore être ajoutés ou validés (Tracked/ Untracked).

NB: Les fichiers non ajoutés ou modifier sont en rouge.

10. Workflow de l'étape 6 a 11:

Premièrement j'ai créer un fichier README.md au niveau du dossier git_test. Deuxièment j'ai taper la commande git status pour bien visualiser que le fichier README.md est pas encore validés (Untracked). Troisièment j'ai taper la commande git add README.md pour ajouter ce dernier, après j'ai taper la commande git commit – message (-m = --message) pour valider les modifications apportées. Finalement j'ai retaper la commande git status et j'ai remarquer que mon fichier a été bien validés (Tracked).

11. M'a donner l'auteur de ce commit (username <email>), la date et qu'est ce qu'il a fait précisement.

```
redkhay@vm:~/Git/test_git$ git log
commit 3a8da490895132554cb9320cdc96350ae06db727 (HEAD -> main)
Author: RedaMeziouni <meziounidb@gmail.com>
Date: Fri Jan 8 00:47:38 2021 +0100

Ajouter README.md
redkhay@vm:~/Git/test_git$
```

Après avoir créé plusieurs commits ou si vous avez cloné un dépôt ayant un historique de commits, vous souhaitez probablement revoir le fil des évènements précédent. Pour ce faire, il suffit d'exécuter la commande git log.

12.

```
redkhay@vm:~/Git/test_git$ touch hello_world.txt
redkhay@vm:~/Git/test_git$ git status
On branch main
Untracked files:
   (use "git add <file>..." to include in what will be committed)
        hello_world.txt

nothing added to commit but untracked files present (use "git add" to track)
redkhay@vm:~/Git/test_git$ |
```

13.

La difference entre « git add README.md » et « git add . », c'est que la premiere ajoute justement le fichier README.md et la deuxieme ajoute tout les fichiers qu'il existe au niveau de notre repertoire. Pour notre cas ca va ajouter README.md et hello world.txt.

```
redkhay@vm:~/Git/test_git$ git commit -m "1st commit"
[main af8ecb1] 1st commit
2 files changed, 4 insertions(+)
create mode 100644 hello_world.txt
redkhay@vm:~/Git/test_git$ |
```

16. Vérifier le statut et faire pousser les fichiers au niveau du remote repo grâce a la commande git push.

```
redkhay@vm:~/Git/test_git$ git status
On branch main
nothing to commit, working tree clean
redkhay@vm:~/Git/test_git$ git push origin main
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (6/6), 521 bytes | 260.00 KiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:RedaMeziouni/test_git.git
* [new branch] main -> main
redkhay@vm:~/Git/test_git$ |
```

17.

```
redkhay@vm:~/Git/test_git$ git status
On branch main
nothing to commit, working tree clean
redkhay@vm:~/Git/test_git$
```

